

HOW IT WORKS

TCP/IP Ethernet Networking

Setting up a network is a piece of cake these days, which is remarkable when you consider how many technologies are involved. Tom Royal peels away the layers

Today's computer operating systems have been designed to make connecting several computers to a network extremely simple. Windows XP and Vista allow you to connect to a standard home Ethernet network with just a few clicks. This makes it easy to forget the complexity of the task that networks undertake, shuttling data between many computers all round the world. This month we'll explain the technology behind TCP/IP Ethernet networking, which is by far the most common standard.

BACK TO BASICS

Although we often refer to an Ethernet or WiFi network, the reality is that even the simplest home network relies on several interconnected technologies. To explain how they work together, they are usually divided into groups known as layers. The most common system, the five-layer TCP/IP model, defines five layers: physical, data link, internet, transport and application.

The best way to think of this model is as a stack with the application layer on top and the physical layer at the bottom. When you use your PC to connect to a website, for example, your request is passed from the web browser to the application-layer technology. The application layer uses the transport-layer technology, which in turn uses the internet-layer technology, and so on down to the physical-layer technology. When a message is received from the network it passes in the opposite direction, moving from the physical layer to the application layer. We'll examine each layer in turn, starting at the top.

APPLICATION FORM

The application layer consists of protocols that allow applications on your PC to interface with the network. Several different application-level protocols can be used on a single network connection simultaneously, adding to the traffic that coexists in the lower levels.

When you switch on a computer connected to a home network the first application-level protocol to be used will probably be Dynamic Host Configuration Protocol (DHCP). This allows your computer to request a network address (IP address) from a server. In a home network the DHCP server is usually built into a router, and as DHCP is usually enabled by default it lets users set up a network without ever having to worry about network addresses. Once Windows has used the DHCP protocol to get an address, your computer's web browser might use DNS, which looks up the numerical IP address of named websites, and then HyperText Transfer Protocol, or HTTP, which is used to request web pages. All

application-layer protocols access the network by issuing requests to the next layer down, the transport layer.

TRANSPORT IS ARRANGED

Application-level protocols rely on the network being able to exchange data with a specific remote computer system. Transport-layer technologies, of which the most common are TCP and UDP, provide this kind of tunnel-like connection. Transmission Control Protocol, or TCP, is the most common. TCP is a key part of the TCP/IP networking system that supports the internet and almost all corporate and home networks.

If you use an application that needs to send data to a computer on the internet, such as a web browser, it passes data to the operating system's TCP system. The TCP system then sets up a data connection with a remote computer. First it asks the remote computer to open a port to receive data. Ports are numbered, and provide a means of distinguishing between different types of data connection. HTTP messages are usually sent over port 80, for example, and the SMTP service used to send email uses port 25. This system allows a single server to differentiate between, for example, messages intended for its web server, which will listen on port 80, and those for an FTP server listening on port 21.

As well as helping to organise the data connections required by different application-layer protocols, TCP plays a vital role in making sure the data sent between computers is received correctly. It receives data from the application layer and packages it into chunks known as segments. Each segment has a header that includes the port on the remote computer that needs to receive the information stored in it. When communicating with the remote computer to arrange an open port to receive information, the TCP system receives a random value known as the Initial Sequence Number, or ISN. The computer then increments this value, and inserts the result into the header of the first sequence. If it sends four sequences of data, for example, the first may have a header containing the number 201. If the receiving computer gets this and the three that follow, its TCP system sends back the next number in sequence: 205. If a sequence is lost along the way it sends back 204, and the sending computer knows it must resend the data. A more advanced error checking system known as selective acknowledgement or SACK is commonly used to enable more precise requests for missing data.

TCP's system for identifying and re-sending missing data makes it ideal for most uses, but in

certain applications it's more important to get most of the data to its destination quickly than to get it all there more slowly. In these instances an alternative system called User Datagram Protocol, or UDP, is often used. Unlike TCP, UDP doesn't keep track of connections. Instead each chunk of data is treated individually, and no effort is made to track missing data. This may not sound very useful, but in applications such as Voice over IP (VoIP) UDP can be very handy. As transport-level protocols, TCP and UDP have one key feature in common: both rely on Internet Protocol, or IP, to send and receive the messages they create and interpret. In the TCP/IP layer model, IP is known as the Internet Layer.

IT'S A SERIES OF TUBES

Like TCP, IP protocol is handled by the operating system of modern computers. Its function is to provide addresses for computers and a method of addressing and sending data between them. IP simply addresses packets of data and sends them without making any effort to ensure correct delivery, as TCP can handle this if needed. IP packages the information passed to it by TCP into bundles called packets. Each packet includes a source address, explaining where the packet has come from, and a destination address.

In order to include this information, IP relies on each computer having a numeric IP address. In the most common IP version, IPv4, an IP address consists of four bytes, or 32 binary bits, of information. These are usually represented numerically, with each byte separated by a dot, in what is known as a dot-decimal address. This gives a total range of IP addresses from 0.0.0.0 to 255.255.255.255.

Of course, the number of addresses available using the IPv4 standard is limited and may run out one day. For this reason a newer standard, IPv6, has been developed. IPv6 uses 128 bits of data per address, providing a total of 2^{128} addresses compared with the 2^{32} available using IPv4. Windows Vista supports IPv6, and can connect to IPv4 and IPv6 networks at the same time, but takeup of IPv6 has been slow. The main reason for this is the popularity of Network Address Translation, or NAT.

We explained NAT in detail back in *Shopper*, December 2004, but in simple terms it lets you connect two networks together. This means you can build your own network using one set of IP addresses, and still connect to resources on the internet. On a home network, NAT is performed by a router. The router has a standard private IP address such as 192.168.0.1, and runs a DHCP server, giving computers on your network similar

addresses. Your notebook computer, for example, might have the address 192.168.0.2, and your desktop PC might have the address 192.168.0.3.

To connect to the internet the router needs an IP address on that network, which is assigned by your ISP. We'll use 230.123.1.23 as an example. The router performs NAT to join this IP address to its local address of 192.168.0.1. If your notebook computer requests an MP3 file located on the desktop computer, it will send a packet of data to the desktop's address of 192.168.0.3. The router can see that this address and the source address of the packet share the same host network address (192.168.0). This tells it that the destination is on the local network, so rather than relaying the packet to the internet it sends the packet to all local network connections. If your notebook computer sends a request for the web page at *www.google.com*, or 209.85.135.147, the router sees that this address is part of a different host network (209.95.135). It will therefore send this packet out to the internet over its WAN connection.

Before doing so, though, NAT needs to make sure that the remote server at *www.google.com* knows where to send replies. The notebook's address of 192.168.0.2 won't mean anything to it, so NAT changes the source address of the packet to the router's own internet address of 230.123.1.23. The router records the change in a table, associating the remote address of 209.85.135.147 with the local address 192.168.0.2. When the router receives a reply from the web server it refers to this table and works out that the page information was requested by the notebook, allowing it to change the destination IP address to 192.168.0.2 and send the packet on to the local network.

ARPING ON

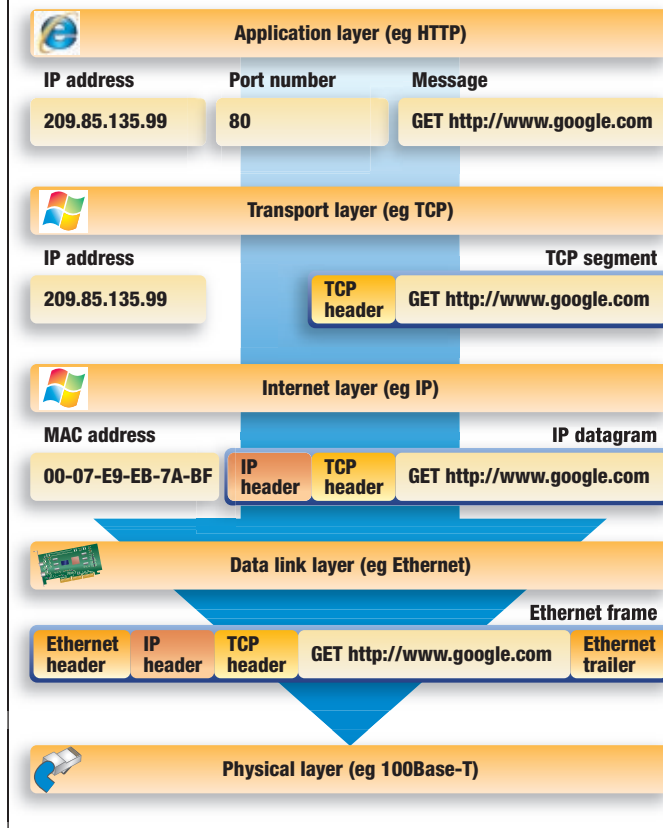
Another important part of the Internet Layer is the Address Resolution Protocol, or ARP for short. ARP translates IP addresses into Media Access Control, or MAC, addresses. Each computer network adaptor has a MAC address permanently programmed into it like a serial number. To convert an IP address into a MAC address, ARP sends out a packet of data containing its own IP and MAC address, as well as the desired IP address and a string of zeroes representing the unknown but wanted MAC address. When the computer using the desired IP address receives this packet it returns a similar one, but including its own MAC address and directed at the source computer.

The ARP process is significantly accelerated by the use of a technique called ARP Caching. Because the outgoing ARP requests don't have a valid destination, any intermediate computer on the network can read them. Even if these computers don't need to respond, they can note the source computer's IP and MAC address in case they need to contact that IP address later.

ETHER SONG

The packets that Internet Protocol creates are moved from place to place by several methods known as the data link layer. The most common data link layer protocols for home and office use are Ethernet, which uses cables to transmit data, and WiFi, which uses radio waves. As Ethernet and WiFi can carry the same TCP/IP traffic, they can work together almost seamlessly, so webpage requests made from your notebook computer can pass through a wireless network

Layer cake The five stages of TCP/IP networking



TCP/IP network activity begins at the application layer with a program such as Internet Explorer sending a message to a specific IP address on a specific port. The Transport layer manages the connection but relies on the Internet layer to actually send and receive the message which is packaged as an IP datagram. The packets that the Internet layer creates are moved from place to place by the Data link layer. This can use a range of different cables, known as the Physical layer, 100Base-T being the most common.

and then, via an access point or router, through wired Ethernet networks to the web server.

Ethernet's principle function is to enable many computers to communicate using a single medium without interfering with one another. It does this by packing the information passed from the internet layer into chunks known as frames. Like TCP and IP, it adds header information to the data. The Ethernet header includes source and destination MAC addresses and a value that identifies the type of content carried inside. In most cases this value will be 0x0800, which signifies an IP packet. The finished frame containing application-layer data, TCP information, IP information and the Ethernet header and footer is finally ready to transmit across the network cable.

In its original, simplest form, an Ethernet network uses a single cable to attach a number of computers, known as nodes. Each node has a network adaptor with a unique MAC address, which it uses to listen to data passing down the cable. When it detects a data frame whose recipient address matches its own MAC, it reads the frame data. When it wants to send a frame, it listens to the cable until it is free of traffic, sends the frame and then continues to listen. If it hears no noise on the cable, it considers the frame sent. It's possible that two computers might transmit at the same time. This is known as a collision, and it results in unusable data. When this happens, the computer sends a jamming signal, which warns all computers that a collision has occurred. After jamming the cable, the two computers that transmitted simultaneously wait for a random period of time before transmitting again. This system, known as Carrier Sense Multiple Access with Collision Detection (CSMA/CD). As the number of nodes on a network increases, the number of collisions

increases. With too many nodes, the number of collisions can make the network too slow for any practical use.

The other main problems with a single cable is that it's limited to 185m in length (unless a signal repeater is used), and if the cable breaks the whole network goes down. Hubs were introduced to combat these problems. With one port per PC, each computer had its own physical connection, so there was no single point of failure and each PC could be 185m from the hub. Hubs use shared bandwidth and are prone to collisions, so networks were split into groups of smaller number of nodes (segments), joined by bridges. A bridge monitors network traffic and shares only the data that needs to be shared between segments.

Today, switches are used, which also provide one port per computer like a hub. However, a switch doesn't use shared bandwidth and passes frames from one node directly to another.

The name Ethernet is commonly associated with eight-wire cables often referred to as RJ45 or Cat5. In reality, Ethernet can run over several different types of cable. The type of cabling is defined not in the data link layer, but in the lowest layer, known as the physical layer. The most common types of physical Ethernet systems used in homes and offices are 10BASE-T, 100BASE-T and 1000BASE-T, also known as Ethernet, Fast Ethernet and Gigabit Ethernet. All three use the same type of standard cable, which includes four twisted pairs of wires. Each pair is tightly wound around itself down the length of the cable to reduce interference. Each cable terminates in a plastic plug known as an 8P8C connector, although these are often referred to incorrectly as RJ45 plugs. 10BASE-T and 100BASE-T use only two of the wire pairs in such a cable, but 1000BASE-T uses all four.